Towards a Software Framework for Automatic Business Process Redesign

Marwa M.Essam[1], Selma Limam Mansar[2]
[1] Faculty of Information and Computer Sciences –Ain Shams University/Cairo, Egypt
Email: marwa.essam@gmail.com
[2] Carnegie Mellon University-Qatar /Information Systems Program, Doha, Qatar
Email: selmal@qatar.cmu.edu

Abstract— **A key element to the success of any organization is the ability to continuously improve its business process performance. Efficient Business Process Redesign (BPR) methodologies are needed to allow organizations to face the changing business conditions. For a long time, practices for BPR were done case-by-case and were based on the insights and knowledge of an expert to the organization. It can be argued that efficiency, however, can further be achieved with the support of automatic process redesign tools which are few at the moment. Process mining as a recent approach allows for the extraction of information from event logs recorded in different information systems. In this paper we argue that results driven by process mining techniques can be used to capture the various types of inefficiencies in the organization and hence propose efficient redesigns of its business model. We first give an outline on the current directions towards automatic BPR followed by a review on the different process mining techniques and its usage in different applications. Then, a specific framework of a Software tool that uses process mining to support automatic BPR is presented.**

Index Terms— **Process Mining, Business Process Redesign, Business Process Management**

I. Introduction

A business process is a collection of related, structured activities that produce a service or product that meets the needs of a client. Business processes are critical to any organization as they generate revenue and often represent a significant proportion of costs. Nowadays, many Business Process Management (BPM) systems exist in the market (Ex.: FileNet and Ultimas). BPM systems provide organizations with a broad range of facilities to design, enact, control and analyze their business process [1]. A list of the some of the cross-industry BPM suites with their relative strengths can be found in [2].

Despite its popularity and obvious pay-offs, the current practices for monitoring and analyzing the execution of BPM systems in the organizational reality still leaves a lot to be desired [3]. There is a vital need for BPM systems to satisfactory support Business Process Redesign (BPR). In many cases, the developed business functions do not effectively reflect the actual business process. Many of the implemented business functions are never used. Other business functions provide more functionality than actually needed. Another issue is related to the evolution of business processes and their variability. In many domains (ex: healthcare), frequent process changes requires the continuous adaptation of the supporting IS.

Currently, most BPR projects depend mainly on an expert to an organization. Experts/ Consultants may use some tools for process modeling, business planning or process prototyping [4]. However, there is currently no tool that supports the automatic redesign from the old business process to new innovative business processes.

In recent years, process mining was introduced in the context of business process management [5]. Process Mining, similarly to data mining, allows for the extraction of information from event logs recorded in BPM-based systems. Some of the possibilities offered by process mining results are the discovery of new business process models, the checking of the conformance to some prescriptive or descriptive models, or the extension of an initial model with analysis data.

In this paper we argue that an evolutionary redesign to business processes can be reached using results driven by process mining techniques. The evolutionary redesign is based on the application of general best practices or heuristic rules to an existing design. We think that realizing adaptations to business process has become a difficult task to accomplish due to the lack of knowledge to customize the process logic at a sufficient level. However, using process mining, different models can be extracted from the reality logs and various types of inefficiencies in the organization can be captured by analyzing these logs. These results can be used as an input base for a tool that suggests efficient redesigns to the business process, hence, providing consultants and experts with a vision on how to get from the old process to the new process.

This paper is arranged as follows: Section II introduces the notion of modeling a business process with an example on a credit application process. Section III discusses some the related work towards converting old process designs to new designs. In Section IV we give a review on the different process mining algorithms and some of its application areas. In Section V, we discuss our vision on how process mining can be used in a software tool that supports the automatic redesign of business processes.

II. Business Process Modeling

A group of related tasks that together create value for a customer is called a business process. Different modeling languages/techniques can be used to represent different

DOI: 01.IJCOM.02.03.42

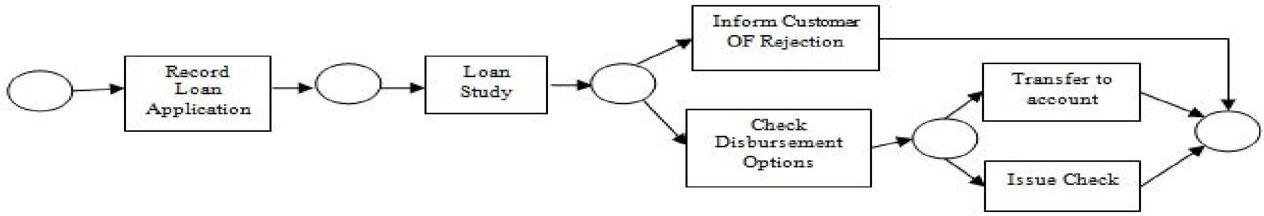

Figure 1. Credit Application Process Model

aspects of the business process. Control Flow modeling languages, like Petri-nets, represent the different activities in the process with constraints to control the execution between them (In what order activities are executed). Data models like Entity Relationship Diagrams represent the data organization in a process. Organizational models like organizational charts represent the structure in which the business process will execute (the entities/roles that can perform work for the enterprise). An Example of a credit application business process is shown in Fig.1. For space limitation we only show the control flow of the process. The process is modeled using Petri-Nets [6]. It begins with the recording of the application where the client expresses an interest in acquiring credit. This stage includes the presentation of the application, and the required documents to the organization for verification. This is followed by an analysis or study of the credit application to decide whether to accept/ reject the credit. The client is notified in case of rejection. In case of acceptance, the credit is disbursed to the client by either a credit transfer to a bank account or by check.

III. Review of Techniques Towards Automatic BPR

A BPR project starts when an organization is faced with the need to change its business process to make improvements in its quality, cost, service, lead-times, outcomes or flexibility. Usually, BPR projects are carried out by setting up workshops within the organization to think of alternatives to the business process. Consultants, employers, managers and specialties participate in these workshops to make process redesigns. Some Software tools may be used within these workshops to aid the redesign process. However, the identification of the problem areas and the opportunities of change are totally determined by the workshop group. As a result to this manual approach, the outcome redesigns are often subjective and non sustainable. This is because it is strongly influenced by the individual expertise of the workshop group and may not suit the actual case of the organization under study.

To make process redesigns that actually succeed on the long term in improving the business process, the research in BPR was recently directed towards making automatic business process redesigns. In [7], a research project was proposed to address the lack of tools in industry with “intelligent” capabilities to suggest favorable alternatives to an existing process design. To develop such an “intelligent” redesign tool, the project proposed the idea of making evolutionary, local updates to an existing workflow design to gradually improve its performance. In this evolutionary approach, the existing process is taken as a starting point and is refined on the basis of what is called “redesign best practices” [8]. A redesign best practice describes a well-tried way to remove a particular problem from a process to improve its performance. An Example of a best practice is: Eliminating unnecessary tasks from a process (the tasks with no added value for customers). In [8], an extensive literature survey has taken place to collect all best practices for evolutionary process improvement.

Towards developing this “intelligent” tool and based on the aforementioned evolutionary approach for BPR, a new technique to find process design alternatives was proposed in [9]. In this technique, a business process first is put in a formal process definition defined by the authors called Proto Net. A set of process measures are then calculated on the process (Ex: Level-of-Control, which is the percentage of control/decision tasks). The authors specified 18 different process measures to be calculated on the process design under study. The calculated measures are then compared against a set of condition statements that when evaluated to true a “redesign” best practice is selected to be applied on the process model (Ex: Apply Task Elimination if level of Control >0.2). Cutoff values for condition statements were determined by the authors’ expertise in the field.

Although this technique suggests the use of some best practices in the process model (the ones that their condition statements evaluated to true), it didn’t specify exactly how these practices will be applied. As pointed out in [10], a redesign best practice just provides directions on how the redesign should be performed. When we look at the parallelism best practice, for instance, it is suggested that the redesign should have more tasks in parallel than the original process. But it does not tell us to put tasks A, B and of process X in parallel. In [10], four exact transformations were suggested to be applied on selected process parts of the process model to produce different redesigns. The input process model is assumed to be a Petri-net with some extensions like data dependencies and roles.

The proposed transformations are:

Unfolding of tasks, in which aggregated tasks (upper level tasks that are modeled in a detailed sub-process) are split up into several smaller tasks.

Parallel Transformation, in which tasks that do not have data dependencies are executed in parallel.

Sequence Transformation, in which all parallel paths in a model are transformed into sequence path, provided that the output sequence will have the lowest throughput time and will contain no errors related to data dependencies.

Merging transformation, in which a task cluster executed by

DOI: 01.IJCOM.02.03.42

the same role is merged into one aggregated task. For each of the above transformations, the authors specified some formal characteristics that the selected process part should have in order to be eligible for the transformation (Ex: A selected part for the parallel transformation must not have a selective route to prevent contradictions with the selection's original purpose). They specified also how exactly the transformed part will be replaced in the Petri net model (Ex: Removing unnecessary edges/tasks). However, still their technique doesn't allow the automatic selection of the process part to be altered or the automatic selection of the transformation that produces the best result.

In [11], an algorithm called the Boolean Verification Algorithm (BVA) was presented for the optimization of workflows. On the contrary to the techniques mentioned before, this approach does not focus on applying best practices to the process design. It just focuses on finding the maximum parallelization for a design to reduce the overall execution time of the process. For this purpose, BVA uses a method called the if-conversion. The main idea of this method is to assign Boolean activation conditions to workflow tasks based on their control flow dependence. While scanning the process model from the start task , BVA assigns Boolean control parameters on different branches and choice nodes (C1, C2,…Cn) and forms a Boolean activation condition for each task (Ex: !C1 & C2 v C1). Tasks are then checked for their control flow dependencies by analyzing their activations conditions and tasks with no control or data dependencies are parallelized.

Although this algorithm promises to ensure a full parallelization of a business process, it still doesn't relate the reality to the design. In other words, it focuses only on the parallelization of tasks while in some cases, a sequential process may be perceived as a simpler process by employees and clients. Since the order of the tasks is fixed in sequential constructs, the execution of the process is done in the most logical way hence reducing errors. Furthermore, the synchronization that is required after the execution of tasks in parallel is not necessary in sequential processes.

Let us note now that all the techniques that we mentioned above don't satisfactorily support the automatic production of process redesigns. Although they provide guidelines on how to apply different transformations on the process model, none of them supports either the automatic selection of the process part to redesign or the best transformation to apply on it.

TABLE I. A Process Log

Case Identifier	Event Identifier
Case 1	Task A
Case 2	Task A
Case 3	Task A
Case 3	Task B
Case 1	Task B
Case 1	Task D
Case 2	Task C
Case 2	Task D
Case 3	Task D

This is due to the lack of knowledge on what causes the low performance within the process. In the next sections, we will show how process mining can be used to enrich the business process model with the required knowledge to satisfactory allow automatic process transformations.

IV. Business Process Mining

A. Introduction

Process Mining allows for the extraction and the analysis of information from event logs recorded in BPM-based systems. Example of event logs are: the audit trails of a workflow management system, the transaction logs of an enterprise resource planning system, or the electronic patient records in a hospital. The two main components of a record in any event log are: the event (task/activity) that was executed and an identification of the particular instance of the process within which the activity was executed (case). More information can also be available in the log (for example, the timestamp of the event, or the performer of the event, the data elements recorded with the event, etc...).

To illustrate the concept of process mining, consider the log information recorded in Table I. This log contains information on three execution cases (1, 2 and 3). Executed events are represented as tasks and are assumed to be recorded in order. When scanning the log, we can detect four different tasks in the process (Tasks A, B, C and D). One can see also that all cases starts by task A and all cases ends by task D. In two cases (1 and 3), task B follows task A in the execution. In case 3, task C follows task A. From this information, we can simply draw, using Petri Nets, the process model that corresponds to this log as in fig. 2.

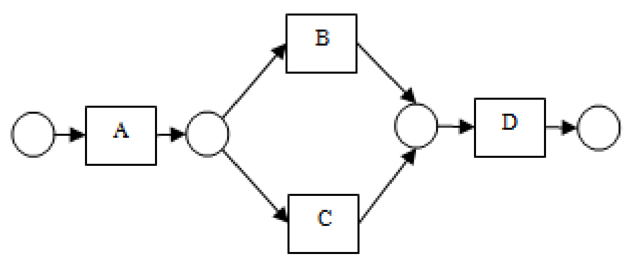

Figure 2. A process model corresponding to the log

B. Process Mining Algorithms

Over the last decade, many process mining algorithms has been developed to find process models that successfully mimics the behavior registered in the logs. In [12], a control flow mining algorithm called the alpha algorithm was presented. The alpha algorithm assumes that there is no noise (logs containing exceptions) in the data and that the log contains sufficient information about the workflow (no paths exist with low probability that prevents them from being detected). The alpha algorithm scans the log and looks for causal relations (a relation between two tasks A and B such that B is directly followed by A in a log trace and A is never followed by B). These causal relations are then represented in a Petri-net model describing the output process.

DOI: 01.IJCOM.02.03.42

To address the issue of noise existing in the log, an algorithm called the Heuristic Miner was presented in [13]. The Heuristic miner follows the alpha algorithm in finding causal relations between tasks. However, it calculates frequencies for the occurrence of each task and for the relation between tasks. Cutoff values are then used to eliminate tasks/ relations that are a result of noise/ un-complete executions. Because in some application areas, the resulting models after process mining become very difficult to understand (Spaghetti-Like), many process mining algorithms based on clustering was introduced to eliminate unnecessary information from the output models. The Fuzzy miner [14] for example starts by drawing a very complicated graph with all causal relations found in the log. It then simplifies the graph by creating clusters of nodes. Clusters are created by aggregating low significant nodes with their highly correlated neighbor nodes. Different significance and correlation matrices can be used for different application purposes. The lion's share of efforts in process mining has been for discovering control flow models resulting in many other models other than the ones mentioned above. However, process mining can also be used to mine much other useful information. As mentioned before, the event logs not only record information about the different cases and the different tasks. It also records information on the role who executed these tasks, the input and output attribute values to and from each task and the execution start and end time of tasks. This information can be used to mine the relations between the different roles creating an organizational model. It can also be used to analyze the information flow between the different roles, the interactions between the co-workers, the decision points in the models and the performance of executions [15], [16].

C. Process Mining Application in BPR

Process mining has been applied in a variety of organizations covering many application domains. In [17], process mining was used to analyze the test process in ASML. ASML makes so-called wafer scanners that are used to manufacture processors in devices ranging from mobile phones to desktop computers. Wafer scanners are really complex machines that use a photographic process to image nanometric circuit patterns onto a silicon wafer. The testing of the manufactured wafer scanners is a time-consuming process. So, the goal of the analysis was to reduce the testing time.

Each wafer scanner in the ASML factory produces a log of the software tests that are executed on it. Process mining was used to visualize the actual flow of the test process and confront this visualization with the idealized view of the tests according to engineers. It was found that as soon as one test fails, a fix is made to the scanner and all other tests are put on hold (idle time) and often after the fix is made, some tests are re-executed again. Visualizing this loop-backs caused by some tests gave the engineers a useful view on what was causing the time loss in the test process. Hence, allowed them to make changes to the test process to reduce the time (for example, execute some tests at earlier phases). Healthcare is another famous application domain for process mining. The applicability of process mining in healthcare was demonstrated using a real case of a gynecological oncology process in the AMC hospital in the Netherlands [18]. The log data contained information about a representative group of 627 gynecological oncology patients. The goal of using process mining was to discover the care paths followed by individual patients and whether certain procedures are followed or not. After applying process mining techniques, many useful results became visible to the people at the hospital. For Example, it was found that patients who undergo several chemotherapy sessions often need to visit the dietician. This was not immediately clear to everyone and illustrates the value of creating transparency using process mining.

The above two mentioned projects were implemented with the process mining tool named ProM [19]. ProM contains more than 250 plug-ins that implement different process mining algorithms. However, it is not clear how to use ProM in process redesign projects. In the above two projects, the authors used different plug-ins but viewed each plug-in result alone. Although ProM allows the results from some algorithms to be integrated in a Colored Petri Net (CPN) that support analysis and simulation, there was no guidance from ProM on how to improve the business processes. Instead, the researchers concluded the redesign ideas from viewing the simulated models. i.e. It is hard to make process redesign using process mining a repeatable service.

V. Proposed Software Framework for Automatic BPR

A. Introduction

Based on the discussion in the previous sections, we will now focus on two phases in BPR that, up to now, are done manually by the designers:

- The designer of the new process manually selects the process part to be redesigned from the old process model. He also decides what change to be made on the selected part.
- When using process mining for BPR, experts/ researchers in process mining determine which process mining algorithm to use and results after modeling do not suggest redesign ideas.

For the above two points, we present our view on a software that automatically outputs a specific redesign to a business process using its recorded log information as input. In Fig. 3, the framework for this software is presented. The proposed software framework is composed of two main parts, a business process miner and a redesign engine. A Business Process miner is responsible simply for applying process mining techniques on the input process log to gather information that will aid the redesign process. The Redesign engine is responsible of generating new redesigns based on both process mining results and knowledge of the redesign best practices. Next, we illustrate both components in more detail.

DOI: 01.IJCOM.02.03. 42

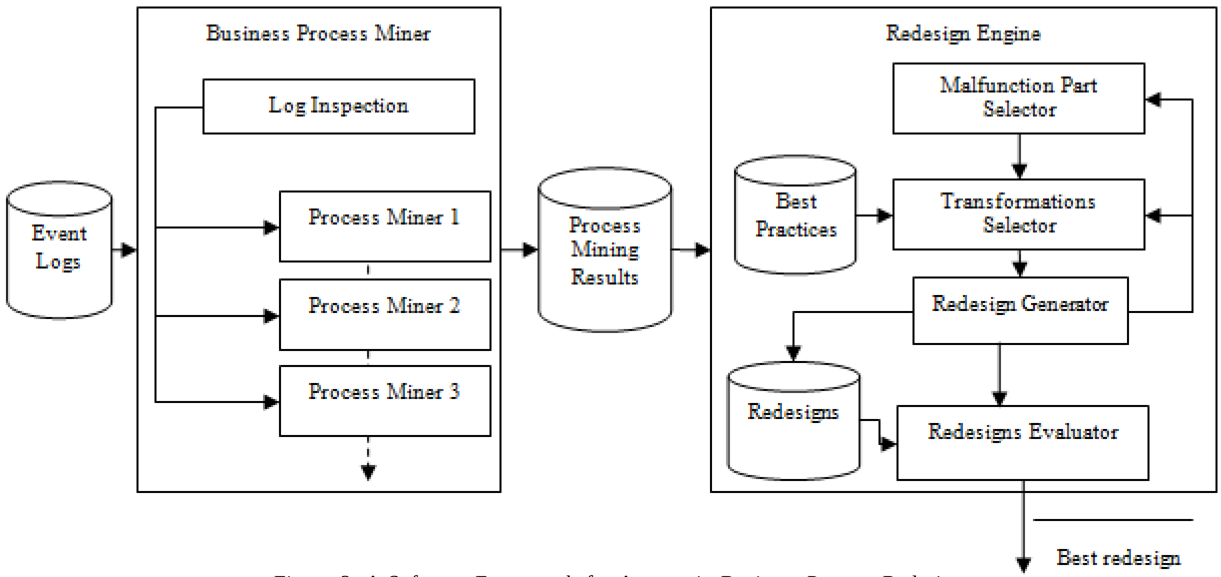

Figure 3. A Software Framework for Automatic Business Process Redesign

B. The Business Process Miner Component

In the business process miner, a proper selection of process mining algorithms is applied on the event logs to produce different model types. All the results gathered are then stored in a process mining results database. The choice of which process mining algorithms to apply on the log will be determined in a log inspection phase using some parametric characteristics from the log (for example, a log which contains a large no of tasks such that mining all these tasks will produce a spaghetti like model will be mined using the fuzzy miner). If role information is available in the log, the log will be mined to produce organizational and social network models. This role analysis is important since in some cases, the malfunction in a process design is mainly because of bad organizational or social structures. The log will be mined also to produce performance data like the throughput time of cases, the slowest tasks, the delays before tasks execution, the resources utilizations,…etc.

C. The Redesign Engine Component

Using results gathered from the business process miner, the redesign engine starts by determining the malfunction part in the process design (a certain component in the process where the mining results show that it somehow causes low performance). This malfunction part selector is the key element in this proposed software as it integrates the results from all the process mining algorithms to come out with conclusions on what causes the low performance of the process (For example, a specific path in the control flow model, a bad organization structure, etc…) . It is important to note also that, to select the problem areas in the business process, the targets for the redesign must be specifically input to the software in order. Targets can be for example, lowering cost, increasing quality, increasing flexibility, etc... The order of targets is important because, in some times targets contradict. For example, increasing the quality in some processes may result in increasing the cost and so on. To find the changes to apply on the selected process part, we suggest the use of redesign best practices. Therefore, the proposed software must contain a database of redesign best practices that contains, for each best practice, an execution rule that shows:

- The conditions that a process part must match in order to be eligible to apply the best practice on,
- And, the process transformation that will be applied if these conditions are true.

Matching the selected process part against the conditions in the best practices data base allows the redesign engine to find what transformations to apply on it. The selected part is then transformed and integrated with the process model in a redesign generator element. Since different combinations of transformations may applied to the same process part (more than one best practice rule evaluate to true), different redesigns to the process model may be generated. Hence, they are saved in a redesigns database. Moreover, if there are more process parts that need to be changed, a redesign loopback starts again by selecting another process part to change. The generated redesigns are then evaluated for the selection of the best redesign. The evaluation will be based on simulating the different redesigns using data from the logs (for example, the arrival time of different cases, the routing probabilities of different paths in the model, the response time from certain roles,… etc).

VI. Conclusions

In this paper we focused on how to automatically redesign business processes in order to increase its performance. We showed that current redesign methodologies let the designer choose the process part to be redesigned and also choose the proper change to apply on it. To allow automatic process redesign, process mining can be used. We briefly presented the concept of process mining and we showed that although a lot of process mining algorithms exist and some were already used in redesign projects, it is still not clear how to make process redesign using process mining a repeatable service. For this reason we presented a framework of a software that

DOI: 01.IJCOM.02.03.42

automatically produces a redesigned business process model using event logs from the old system execution. The basic idea of this software is to integrate the results after using a proper selection of process mining algorithms, and then use these results to select a malfunction process part and transform it using redesign best practices.

The proposed framework provides guidelines on how to build the redesign software. To actually build this software, we plan to work on the following points:

- The different parameters and parameter values that determine the process mining algorithms to use.
- How to integrate the different results from process mining to find the process part to redesign.
- The rules that guide the use of a specific best practice (transformation) on a process part.
- The log analysis parameters that will be used for redesigns simulation.

Acknowledgment

The authors would like to acknowledge that the work for this paper was partly funded by the Qatar Foundation for Education, Science and Community Development. The statements made herein are solely the responsibility of the authors and do not reflect any official position by the Qatar Foundation or Carnegie Mellon University.

References

[1] W.M.P. van der Aalst , A. H. M. Ter Hofstede , M. Weske : "Business Process Management: A Survey", Proceedings of the 1st Int. Conference on Business Process Management, Springer Verlag, 2003.

[2] J. Sinur, J. B. Hill: " Magic Quadrant for Business Process Management Suites", Gartner Research, Pub-Date 18-Oct 2010, ID Number: G00205212

[3] B. Mutschler, M. Reichert, J. Bumiller: "Unleashing the Effectiveness of Process-Oriented Information Systems: Problem Analysis, Critical Success Factors, and Implications", IEEE Trans. on Systems, Man., and Cybernetics, vol.38, no.3, pp. 280 -291, 2008

[4] M. Attaran: "Exploring the Relationship between Information Technology and Business Process Reengineering. Information & Management", 41(5), pp. 585-596, 2004.

[5] W.M.P. van der Aalst , A.J.M.M. Weijters: "Process Mining, a Research Agenda", Special Issue of Computers in Industry, vol.53, no.3, Elsevier Science Publishers, Amsterdam, 2004.

[6] W.M.P. van der Aalst: "The Application of Petri Nets to Workow Management" The Journal of Circuits, Systems and Computers, 1998.

[7] M. Netjes , I. Vanderfeesten, , and H. Reijers: "Intelligent tools for workflow process redesign: A research agenda". Vol. 3812 of Lecture Notes in Computer Science, pp. 444–453. Springer Verlag, Berlin, 2005

[8] H. Reijers, S. L. Mansar: "Best practices in business process redesign: An overview and qualitative evaluation of successful redesign heuristics" Omega: The Int. Journal of Management Science, 33(4):283–306, 2005

[9] M. Netjes, S. L. Mansar, H.A. Reijers, W.M.P. van der Aalst:" An Evolutionary Approach for Business Process Redesign: Towards an Intelligent System", Proceedings of the 9th Int. Conference on Enterprise Information Systems. pp. 484–494. INSTICC, Setubal (2007)

[10] M. Netjes, H.A. Reijers, and W.M.P. Aaslt: "On the Formal Generation of Process Redesigns", First Int. Workshop on Model Driven Engineering for Business Process Management, pp. 49–60, 2008,

[11] A. A. Kalenkova: "Application of If-Conversion to Verification and Optimization of Workflows", Programming and Computer Software, vol. 36, no. 5, pp. 276–288, 2010.

[12] W.M.P. van der Aalst, A.J.M.M. Weijters, L. Maruster. "Workflow Mining: Discovering Process Models from Event Logs", IEEE Transactions on Knowledge and Data Engineering, 16(9):1128–1142, 2004.

[13] A.J.M.M. Weijters, W.M.P. van der Aalst. "Rediscovering Workflow Models from Event-Based Data using Little Thumb", Integrated Computer-Aided Engineering, Vol. 10 (2003)

[14] C.W. Günther, W.M.P. van der Aalst, "Fuzzy mining - adaptive process simplification based on multi-perspective metrics". Proceedings of the 5th Int. Conference on Business Process Management, Brisbane, Australia, 2007

[15] M. Song,, W.M.P. van der Aalst. " Towards Comprehensive Support for Organizational Mining." Decision Support Systems 46(1), pp. 300-317 ,2008.

[16] A. Rozinat, W.M.P. van der Aalst. "Decision_Mining_in Business_Processes". BPMCenter.org Report, 2006.

[17] A. Rozinat, I. S. M. de Jong, C. W. G¨unther, W. M. P. van der Aalst, "Process Mining Applied to the Test Process of Wafer Scanners in ASML,", IEEE Trans. on: Systems, Man, and Cybernetics, vol.:39, no.:4, pp: 474 - 479, 2009

[18] R.S. Mans, M.H. Schonenberg, M. Song, W.M.P. van der Aalst, P.J.M. Bakker, "Application of Process Mining in Healthcare – A Case Study in a Dutch Hospital", *Communications in Computer and Information Science, 25, pp. 425-438*). Berlin: Springer,2008.

[19] The ProM tool, http://prom.win.tue.nl/tools/prom/

DOI: 01.IJCOM.02.03.42